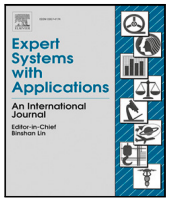
Expert Systems With Applications 248 (2024) 123478

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



GAP: A novel Generative context-Aware Prompt-tuning method for relation extraction

Zhenbin Chen [a], Zhixin Li [a,*], Yufei Zeng [a], Canlong Zhang [a], Huifang Ma [b]

[a] Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin 541004, China
[b] College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

ARTICLE INFO

Keywords:
Relation extraction
Prompt-tuning
Pretrained language model
Few-shot learning
Contrastive learning

ABSTRACT

Prompt-tuning was proposed to bridge the gap between pretraining and downstream tasks, and it has achieved promising results in Relation Extraction (RE). Although the existing prompt-based RE methods have outperformed the methods based on fine-tuning paradigm, these methods require domain experts to design prompt templates, making them hard to be generalized. In this paper, we propose a **G**enerative context-**A**ware **P**rompt-tuning method (**GAP**) to address these limitations. Our method consists of three crucial modules: (1) a pretrained prompt generator module that extracts or generates the relation triggers from the context and embeds them into the prompt tokens, (2) an in-domain adaptive pretraining module that further trains the Pretrained Language Models (PLMs) to promote the adaptability of the model, and (3) a joint contrastive loss that prevents PLMs from generating unrelated content and optimizes our model more effectively. We observe that the context-enhanced prompt tokens generated by GAP can better guide PLMs to make more accurate predictions. And the in-domain pretraining can effectively inject domain knowledge to enhance the robustness of the model. We conduct experiments on four public RE datasets with supervised and few-shot settings. The experimental results have demonstrated the superiority of GAP over existing benchmark methods and GAP shows remarkable improvements in few-shot settings, with average F1 score enhancements of 3.5%, 2.7%, and 3.4% on the TACRED, TACREV, and Re-TACRED datasets, respectively. Furthermore, GAP still achieved state-of-the-art (SOTA) performance in supervised settings.

1. Introduction

With the advancement of computer networking technologies (Dong et al., 2023; Mohajer, Daliri et al., 2022) such as edge computing (Mohajer, Sorouri et al., 2022), the volume of data is increasing rapidly, making it increasingly challenging to manually extract knowledge from those unstructured data. Relation extraction is a fundamental task in natural language processing (NLP) and information retrieval. It plays a crucial role in transforming unstructured text into structured knowledge. Extracting relation triples from unstructured data is of significant importance for many downstream tasks, including knowledge graph construction (Lin, Liu, Sun, Liu, & Zhu, 2015), question answering systems (Zhong et al., 2022), information retrieval (Han, Zhao, Ding, Liu, & Sun, 2022; Lee, Seo, & Choi, 2019), and sentiment analysis (Zeng, Li, Chen & Ma, 2023; Zeng, Li, Tang, Chen and Ma, 2023), among others. The advancement in the field of relation extraction can indirectly drive progress in intelligent systems, further providing support for various industries in society. Relation extraction aims to identify and classify the associations or connections between the entity pairs mentioned in a text. For instance, in the sentence "Barack Obama was born in Honolulu", relation extraction aims to recognize the relation "born in" between the entities "Barack Obama" and "Honolulu", as shown in Fig. 1. However, relation extraction is still suffering from the noise propagation and data sparsity issues.

With the advancement of pre-trained language model technology, the methods based on fine-tuning paradigm is currently the mainstream solution, leveraging the robust natural language understanding capabilities of Pretrained Language Models (PLMs). Most Transformer-based pretrained language models such as BERT (Devlin, Chang, Lee, & Toutanova, 2018), Roberta (Liu et al., 2019) are trained on large-scale text corpora and employ Masked Language Model (MLM) objective for pretraining. The core idea of fine-tuning paradigm is to add a downstream relation classifier on top of the pretrained language models and further update the model's parameters using annotated data to make

* Corresponding author.
E-mail addresses: chenzb@stu.gxnu.edu.cn (Z. Chen), lizx@gxnu.edu.cn (Z. Li), zengyf@stu.gxnu.edu.cn (Y. Zeng), clzhang@gxnu.edu.cn (C. Zhang), mahuifang@nwnu.edu.cn (H. Ma).

https://doi.org/10.1016/j.eswa.2024.123478
Received 29 January 2023; Received in revised form 20 December 2023; Accepted 11 February 2024
Available online 13 February 2024
0957-4174/© 2024 Elsevier Ltd. All rights reserved.

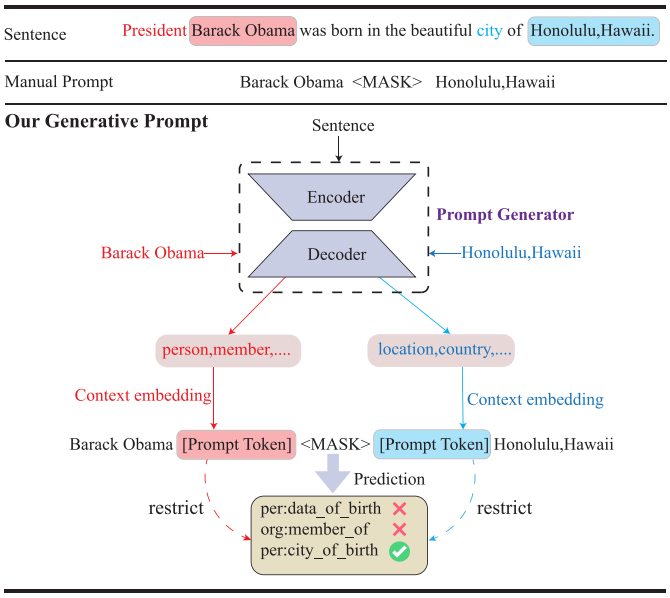


Fig. 1. The core idea of GAP construction. GAP takes both the entity and the sentence as inputs into the Prompt Generator. The Prompt Generator extracts trigger words related to the current entity from the sentence and embeds these trigger words into prompt tokens using contextual embeddings for prompt-based relation extraction.

it more suitable for the relation extraction task. In recent years, relation extraction methods based on fine-tuning paradigm have achieved satisfactory results (Joshi et al., 2020; Li, Sun et al., 2021; Nan, Guo, Sekulić, & Lu, 2020; Soares, FitzGerald, Ling, & Kwiatkowski, 2019; Tang et al., 2020; Wang, Focke, Sylvester, Mishra, & Wang, 2019). Furthermore, it is worth noting that existing methods have demonstrated that not all the words in the text are beneficial to RE (Zhang et al., 2021). Only the context related to the entity pairs or trigger words is crucial for PLMs to perform accurate relation classification. Hence, the model's performance relies on its ability to capture useful context or trigger words effectively. Besides, the inconsistency between the pretraining objectives of the PLMs and the downstream RE tasks limited the models benefit from the pretraining.

Recently, prompt-tuning (Chen, Liu et al., 2022; Chia, Bing, Poria, & Si, 2022; Han et al., 2022; Li & Liang, 2021; Liu, Ji et al., 2021; Liu, Yuan et al., 2023) was proposed to address this issue. By reformulating the downstream task to a cloze-style task and using PLMs as the predictor, prompt-tuning can bridge the gap between pretraining and downstream tasks. Specifically, prompt-tuning inserts text pieces with "[MASK]" token as the prompt templates and concatenates it with the original input. Next, the concatenated sequence will be input into the PLMs to predict which word should replace the [MASK] token. It then maps the predicted label words to the corresponding target sets. The words predicted by the PLMs will be mapped to the corresponding classification labels through a verbalizer. Taking sentiment analysis task as an example, when we want to predict the sentiment polarity of the sentence "I love this movie" by prompt-tuning, we need to concatenate the prompt template "it's [MASK]!" with the sentence and input them into PLMs for prediction. Assuming that the word predicted by the PLMs is "great", the next step is to map the predicted word "great" to the sentiment polarities set (positive, negative, etc.). As we all know, the word "great" conveys the positive sentiment polarity. As a result, this sentence will ultimately be classified into the positive category. In summary, the construction of prompt templates is a crucial step in prompt learning, and it directly impacts the model's performance.

Prompt-based relation extraction methods (Chen, Zhang et al., 2022; Han et al., 2022) have achieved satisfactory results, but there are still some issues that need to be addressed. **Firstly, template engineering is a time-consuming and labor-intensive task.** Prompt-tuning requires domain experts to design specific prompt templates for the relation extraction datasets from different domains. This implies that we need to invest more time and human resources in template engineering. Although prompt-tuning methods based on automated retrieval have been proposed, such as autoprompt (Shin, Razeghi, Logan, Wallace, & Singh, 2020), they require significant computational resources, making them challenging to be generalized. **Secondly, most existing prompt-tuning based RE methods are incompatible with the traditional Language Model (LM) objective.** The reason is that it is unlikely to see many different sentences with the same prefix in the pre-training corpus. **Thirdly, the context related to the entities or relation labels within sentences is not fully utilized.** While PLMs have strong natural language understanding capabilities, prompt-tuning tends to be more unstable compared to fine-tuning. If there is interfering information in the sentence, the prediction may easily shift. Therefore, it is necessary to utilize effective contextual information to enhance prompts. As shown in Fig. 1, the word *President* in the sentence *"President Barack Obama was born in the beautiful city of Honolulu, Hawaii"* implies that the entity type of *Barack Obama* is a person, while *city* suggests that *Honolulu, Hawaii* is a location. Based on these entity information, we can further determine the type of relationship between them.

To address the aforementioned issues, we propose a **G**enerative context-**A**ware **P**rompt-tuning method (**GAP**) for relation extraction. Our method propose a prompt generator that can be used to obtain the context-aware prompt tokens by extracting or generating trigger words associated with the entities. We embed these contextual cues or trigger words into prompt tokens and concatenate these prompt tokens with the input for downstream relation extraction task. Compared to manually crafted prompt templates, our approach does not require as much time and effort in template engineering, while also addressing the issue of compatibility with the traditional LM objective. To alleviate the issue of instability in the prompt-tuning framework during training, we introduce a novel joint contrastive loss to optimize our model. Furthermore, to improve the adaptability of the prompt-tuning model, we introduce the N-gram masking strategy to further pretrain the PLMs. Our method not only effectively reduces the human effort used for prompt template construction, but also achieves better performance in the RE task. We conduct the experiment on four public RE datasets, which has demonstrated the proposed method outperforms the existing SOTA result in both datasets and experimental settings.

Our contributions can be summarized as follow:

- To address the issue of the gap between pre-training and downstream tasks in existing relation extraction tasks, we propose to use the prompt-tuning architecture for relation extraction. We find that compared to traditional fine-tuning paradigm for relation extraction, prompt-based relation extraction methods make it easier to achieve superior performance.
- We propose a context-aware pretrained prompt generator based on the encoder–decoder architecture for relation extraction, which employs the contextual information for generating effective prompt tokens. To ensure the quality of prompt tokens, we also employ a pointer mechanism that prevents the model from embedding ineffective trigger words into prompt tokens.
- We designe a joint contrastive loss to optimize our model, which uses KL divergence to weight the cross-entropy loss and the contrastive loss. This allows the model to learn better from challenging samples and further enhances the performance of relation extraction. Furthermore, we introduce in-domain pretraining to inject domain knowledge into PLMs, enhancing the domain adaptability of our model.

In addition, experimental results on four public benchmark relation extraction datasets showed that our methods improved over PTR (Han et al., 2022) by 82.7% in F1 score on the TACREV dataset, and there were also improvements in the other datasets.

2. Related work

In this section, we will briefly introduce the work related to relation extraction and the prompt-tuning paradigm. At the end of this section, we will describe how our work differs from previous research.

2.1. Relation extraction

Relation extraction (RE) is a classical and crucial research topic in natural language processing, with the goal of extracting and classifying relation triples < Head Entity, Relation, Tail Entity > from unstructured text.

Conventional models for relation extraction proposed in early years are mainly include feature-based models (Bunescu & Mooney, 2005; Kambhatla, 2004; Suchanek, Ifrim, & Weikum, 2006) and kernel-based models (Mooney & Bunescu, 2005; Qian, Zhou, Kong, Zhu, & Qian, 2008; Zhou, Zhang, Ji, & Zhu, 2007). However, these methods highly depend on manual feature engineering, which leads to issues with error propagation. As a result, the performance of these models is unsatisfactory.

With the rapid development of machine learning techniques, it became the mainstream solution for relation extraction. These methods have alleviated the problem of error propagation and achieved promising results. Liu, Sun, Chao, and Che (2013) first proposed to utilize the convolutional neural network (CNN) for relation extraction. Afterwards, Piecewise Convolutional Neural Networks (PCNN) was proposed by Zeng, Liu, Chen, and Zhao (2015) and achieved better performance. They utilize the sentence-level feature and WordNet hypernyms of entities. On the other hand, the RNN-based approaches also show promising results in learning the linguistic structure in text. Zhang and Wang (2015) adopted the bidirectional recurrent neural network (Bi-RNN) to learn the long-term dependency between two entities. But RNNs are always suffering from the vanishing gradient problem, which makes the training process difficult. To solve this problem, Zhang, Zheng, Hu and Yang (2015) proposed to use the bidirectional Long Short-Term Memory network (Bi-LSTM) and external features to improve the performance of relation extraction. Also, there are lots of works that have modified recurrent neural networks for RE (He et al., 2018; Katiyar & Cardie, 2016; Miwa & Bansal, 2016; She, Wu, Wang, & Chi, 2018; Su et al., 2017; Xu et al., 2015; Zhou et al., 2016).

Attention-based models can efficiently capture useful information in context and have been widely used in various NLP tasks. Shen and Huang (2016) proposed to employ a word-level attention mechanism to obtain the critical information for relation representation and use CNN for relation extraction. Later, the attention mechanism also applied to the LSTM model (Zhou et al., 2016). Xiao and Liu (2016) separated the sentence into three parts according to the position of the entities and proposed a hierarchical recurrent neural network for better performance. Additionally, Lee et al. (2019) obtained state-of-the-art performance by incorporating an entity-aware attention mechanism with a latent entity typing (LET).

Graph-based models are popular for many NLP tasks as they are better at expressing associative relationships between entities. Quirk and Poon (2017) built a graph from the sentences where every word is considered as a node in the graph. Edges are created based on the adjacency of the words, dependency tree relations, and discourse relations. Then, they extract all the paths from the graph starting from head entity to entity tail entity as the features for cross-sentence relation extraction. Rather than using the dependency paths, Song, Zhang, Wang, and Gildea (2018) used an LSTM on a graph. They used a graph LSTM structure which contains not only the word adjacency edges but also the edges from dependency tree for relation extraction. Also, there are lots of other graph-based works (Kipf & Welling, 2016; Peng, Poon, Quirk, Toutanova, & Yih, 2017; Vashishth, Joshi, Prayaga, Bhattacharyya, & Talukdar, 2018) for relation extraction.

Recently, the Pretrained Language Models (PLMs) based on the Transformer have shown impressive performances in various NLP tasks and gradually became the powerful backbones for relation extraction. The Transformer-based methods heavily rely on fine-tuning paradigm. They used the PLMs such as BERT (Devlin et al., 2018) and Roberta (Liu et al., 2019) as backbone, proposed a well-designed classifier for downstream tasks and utilized the datasets to fine-tuning the models. R-BERT (Wu & He, 2019) proposed a novel method that combines pretrained BERT language representation with information from the target entities for improving the performance. On the other hand, many knowledge-enhanced language models were gradually proposed for relation extraction and has shown promising results. SpanBERT (Joshi et al., 2020) applied masking contiguous random spans strategy to pretrain the PLMs or better representations of the entity span. KNOW-BERT (Peters et al., 2019) proposed to use the entity linker to retrieve relevant entity embeddings in external knowledge bases and integrate WordNet information into BERT for getting knowledge-enhanced entities representation. And LUKE (Yamada, Asai, Shindo, Takeda, & Matsumoto, 2020) obtained contextualized representations for words and entities through a masked language model pretraining task on a Wikipedia-based entity-annotated corpus. They used entity-aware self-attention mechanism for achieving impressive performance on relation extraction. Although the methods based on fine-tuning paradigm have achieved promising results, they still have limitations: there is a certain gap between the objectives of the upstream and downstream tasks, which limited their performance.

Prompt-based methods have been proposed to solve these problems and have drawn some attention in recent research. PTR (Han et al., 2022) and KnowPrompt (Chen, Zhang et al., 2022) are the two most well-known prompt-based approaches for relation extraction. PTR (Han et al., 2022) leveraged logic rules to construct prompts with sub-prompts to encode class-specific prior knowledge, which effectively combined human knowledge and pretrained language models in relation extraction task. While KnowPrompt (Chen, Zhang et al., 2022) incorporated entity information into prompt construction using learnable virtual template words and answer words, achieving effective results on relation extraction across various datasets. Also, there are lots of prompt-tuning based works that have been proposed for joint relation extraction (Cabot & Navigli, 2021; Sainz, de Lacalle, Labaka, Barrena, & Agirre, 2021) and uniform extraction (Lu et al., 2022).

2.2. Prompt-tuning

In recent years, large language models (LLMs) have significantly demonstrated capabilities surpassing those of traditional pre-trained language models (PLMs) in natural language understanding and generation. LLMs have become the hot topic of research. And traditional fine-tuning methods are challenging to apply to LLMs with massive parameters. Although prompt-tuning was initially proposed to alleviate the gap between upstream and downstream tasks (Chen, Liu et al., 2022; Chen, Zhang et al., 2022; Chia et al., 2022; Liu, Yuan et al., 2023), researchers have also discovered its effectiveness in efficiently fine-tuning large language models (LLMs). After the GPT-3 was proposed, researchers discovered that by adding a short piece of prompt text before the inputs, such as "Translate English to French:", the model could be directed to perform specific translation tasks (Schick & Schütze, 2020, 2021). The model can rely entirely on the knowledge learned by the model during pretraining to make predictions. There is no need to fine-tune the model using task-specific supervised data in downstream tasks. As a result, prompt-tuning has been applied to various low-resource NLP tasks and has achieved surprisingly impressive performance (Gu, Han, Liu, & Huang, 2021; Shin et al., 2020). PPT (Gu et al., 2021) proposed unified prompt templates for different sentence classification tasks, including sentence-pair classification task, multiple-choice classification task and single-sentence classification task. However, the manual prompt templates require

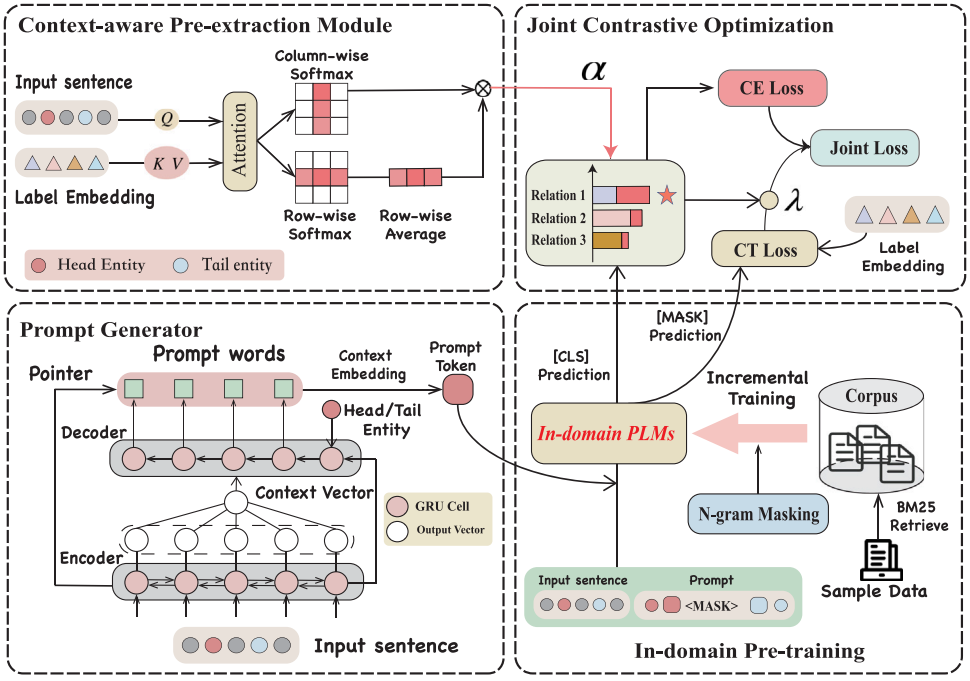


Fig. 2. The model architecture of GAP.α denotes the result of the pre-extraction, which will be added with the prediction result.CE denotes the Cross-Entropy Loss and CT Loss denotes the Contractive Loss designed for GAP.

manual design and it is difficult to find the optimal one, some studies have proposed the methods to automatic construct the prompt templates. Shin et al. (2020) proposed the AutoPrompt which use gradient-guided search to generate the prompt template in the vocabulary automatically. Li and Liang (2021) proposed Prefix-tuning which learn a continuous prompt embedding with fixed PLM parameters, which achieves SOTA results on both text summarization and table-to-text tasks. And Liu, Ji et al. (2021) added the prompt to the input sequence as prefix tokens in different layers further improving the Prefix-tuning.

Certainly, prompt-tuning is also applied to specific NLP tasks. Li, Gao et al. (2021) proposed SentiPrompt for aspect-based sentiment analysis (ABSA) task, which use the sentiment knowledge enhanced prompts to tune the language model in the unified framework. Song et al. (2023) addressed the challenge of modeling label correlations in multi-label text classification and introduced the Label Prompt Multi-label Text Classification model (LP-MTC). Liu, Zhang et al. (2023) proposed a Knowledge Enhanced Prompt Tuning (KEPT) framework for event causality identification (ECI). KEPT incorporated external knowledge from knowledge bases (KBs) to enhance causal reasoning by converting the knowledge into textual descriptions and using interactive and selective attention mechanisms. Chen and Sun (2023) proposed CP-Rec, a conversational recommender system that utilized a contextual prompting framework for dialogue management. CP-Rec optimized prompts based on context, topics, and user profiles. It incorporated a topic controller for subtask planning, a prompt search module for context-aware prompts, and external knowledge for enriched user profiles and knowledge-aware recommendations. Similarly, our GAP leverages contextual information to optimize prompts and introduces additional domain-specific knowledge through in-domain pre-training.

2.3. The differences with other methods

While prompt learning has already been applied to relation extraction in existing methods like PTR (Han et al., 2022) and KnowPrompt (Chen, Zhang et al., 2022), our approach has several distinct differences. Compared to PTR (Han et al., 2022), our approach does not require manual design of prompt templates and logical rules, which makes our approach easier to be generalized. Although KnowPrompt (Chen, Zhang et al., 2022) also employed information-enhanced prompt tokens for relation extraction tasks, there are still many distinctions between our approach and theirs. Firstly, while KnowPrompt (Chen, Zhang et al., 2022) utilized prior information about entities and relation labels, these pieces of prior information are highly randomized within each mini-batch, leading to unstable model predictions. In contrast, our GAP injects prior knowledge based on the context of the current sample and introduces a pre-extraction module to prevent significant model prediction shifts. Secondly, KnowPrompt did not address the issue of domain adaptability, making it challenging to be generalized. Prompt-tuning heavily relies on the external knowledge learned by PLMs during the pre-training phase. If the PLMs lack domain-specific knowledge relevant to the current dataset, model performance can significantly degrade. In contrast, GAP enhances the robustness by automatically searching for and learning relevant knowledge through its in-domain pre-training module. Thirdly, we have designed a novel loss function inspired by contrastive learning ideas to ensure better convergence of our model.

3. Methodology

In this section, we begin to introduce the components of GAP: Firstly, we will introduce the Context-aware Pretrained Prompt Generator in Section 3.2; And then, we will introduce the in-domain adaptive pretraining module in Section 3.3; Finally, we will introduce the Joint Contrastive Loss in Section 3.4. The overview of the method is shown in Fig. 2.

3.1. Task definition

The RE dataset can be denoted as $D = (\mathcal{X}, \mathcal{Y})$, where $\mathcal{X}$ is the set of instance and $\mathcal{Y}$ is the set of relation labels. For instance, $x = \{w_1, w_2, \ldots, w_{n-1}, w_n\}$, the goal of RE is to predict the relation $y \in \mathcal{Y}$ between subject entity $w_{sub} = \{w_s, w_{s+i<n}\}$ and object entity $w_{obj} = \{w_o, w_{o+j<n}\}$, where i and j denoted the length of the subject and object entity.

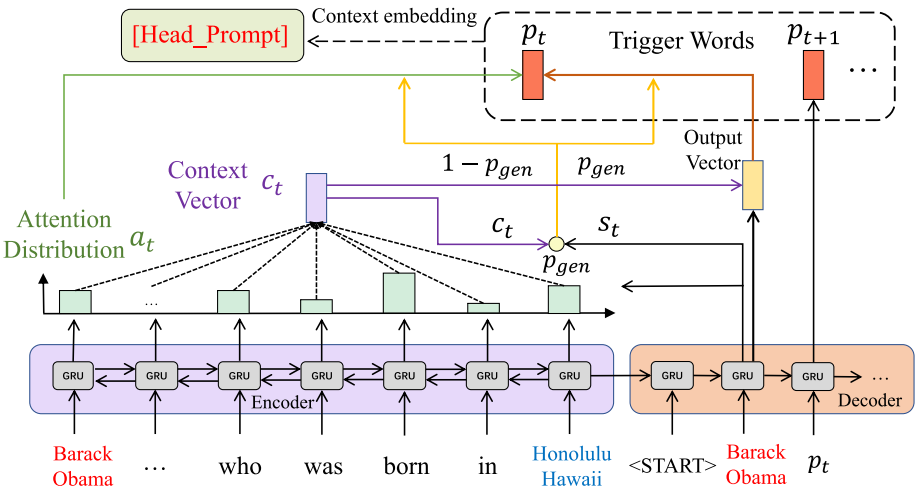


Fig. 3. The model architecture of the pretrained prompt generator.

3.2. Context-aware pretrained prompt generator

We propose to use prompt-tuning to overcome the issue of inconsistency between upstream and downstream tasks in relation extraction, further enhancing its performance. However, there are still two issues that need to be addressed in prompt-based relation extraction methods. Firstly, prompt-based relation extraction methods belong to generative extraction, which makes it less stable compared to traditional extraction methods. Secondly, we need to design suitable prompt templates for relation extraction task to achieve satisfactory results, and the design of prompt templates remains a critical issue in prompt-tuning. In this paper, we introduce the Context-aware Pretrained Prompt Generator to address the aforementioned two issues. The Context-aware Pretrained Prompt Generator consists of two components: a context-aware pre-extractor and a pretrained prompt generator.

3.2.1. Context-aware pre-extractor

To prevent significant shift in prompt-tuning methods when predicting the relations between entities, we have designed a context-aware pre-extractor that does not require training. It employs a simple scaled dot-product attention mechanism to capture the relevance between the global semantic information within sentences and the relation labels, providing an initial judgement of its label. The final relation extraction results will be a synthesis of the predictions from the pre-extractor and those from PLMs. For the given word embedding of the sentence $\mathcal{W} = \{w_1, w_2, \dots, w_{n-1}, w_n\}$ and the embedding of the relation labels $\mathcal{Y} = \{y_1, y_2, \dots, y_{m-1}, y_m\}$, we utilize scaled dot-product attention to calculate the pair-wise matching matrix M from them. Whereas, $\mathcal{W}$ serves as the query, and $\mathcal{Y}$ functions as the key and value and the attention coefficients $M(i, j)$ denotes the relevance coefficients between word w_i and relation y_j.

$$M(i,j) = {w_i}^T \cdot y_j,\ w_i \in x,\ y_j \in \mathcal{Y}, \tag{1}$$

Inspired by Attention Over Attention method (Cui et al., 2017), the *context-to-label relevance matrix* $M_{c2l} \in \mathbb{R}^{|\mathcal{W}|\times|\mathcal{Y}|}$ and the *label-to-context relevance matrix* $M_{l2c} \in \mathbb{R}^{|\mathcal{W}|\times|\mathcal{Y}|}$ can be computed by applying an axis-wise softmax operation on the pair-wise matching matrix M.

$$M_{c2l}^{(t)} = \mathrm{softmax}[M(1,t), \dots, M(|x|,t)], \tag{2}$$

$$M_{l2c}^{(t)} = \mathrm{softmax}(M(t,1), \dots, M(t,|\mathcal{Y}|)), \tag{3}$$

$$M_{c2l} = [M_{c2l}^{(1)}, M_{c2l}^{(2)}, \dots, M_{c2l}^{|\mathcal{Y}|}], \tag{4}$$

$$M_{l2c} = [M_{l2c}^{(1)}, M_{l2c}^{(2)}, \dots, M_{l2c}^{(|x|)}], \tag{5}$$

The *context-to-label relevance matrix* M_{c2l} and the *label-to-word relevance matrix* M_{l2c} denoted the semantic relevance from different view. The *context-to-label relevance matrix* M_{c2l} reflected the relevance between each relation label and the global semantics of the sentences, while the *label-to-word relevance matrix* M_{l2c} reflected the relation tendency presented by each word. Then, the average vector $\beta \in \mathbb{R}^{|\mathcal{Y}|}$ can be computed by applying a column-wise average on the M_{c2l}.

$$\beta = \frac{1}{n}\sum_{t=1}^{|\mathcal{Y}|} M_{c2l}(t), \tag{6}$$

To consider the relevance from both perspectives, we perform a dot product operation between β and M_{l2c}, and then use a softmax function to obtain the final pre-extraction probability distribution $\alpha \in \mathbb{R}^{|\mathcal{Y}|}$.

$$\alpha = \mathrm{softmax}\left(\frac{M_{l2c}^T \cdot \beta}{\sqrt{d_\beta}}\right), \tag{7}$$

where d_β is the dimension of the β.

3.2.2. Pretrained prompt generator

To overcome the challenge of obtaining difficult-to-acquire prompt templates, we propose a pretrained prompt generator to extract or generate entity-related trigger words. These trigger words are then embedded into prompt tokens using context embedding, and these prompt tokens are used as the templates for relation extraction tasks. It has been demonstrated that entity information and trigger words are the most important information for RE in each instance in existing studies (Zhang et al., 2021). Through pilot experiments, we found that enhancing prompt tokens with entity-related context or trigger words effectively improves the performance of relation extraction task. We intende to use an encoder–decoder architecture based on GRU as the prompt generator. However, due to the poor parallel processing capability of GRU, the training become highly unstable. Therefore, we decide to pretrain the prompt generator on a text summarization task to provide it with better initialization parameters. Besides, we introduce a pointer mechanism (Li, Peng, Tang, Zhang and Ma, 2020) to improve the quality of the generated triggers.

Specifically, the embeddings of the sentence $\mathcal{W} = \{w_1, w_2, \dots, w_{n-1}, w_n\}$ are sequentially input into a single-layer bidirectional GRU encoder, producing a sequence of *encoder hidden states* $H = \{h_1, h_2, \dots, h_i, \dots, h_n\}$.

$$H = GRU_{enc}(\mathcal{W}), \tag{8}$$

As illustrated in Fig. 3, in order to obtain the entity-related trigger words or context, we embed the subject and object entities to T_{sub} and

T_{obj} by Eqs. (9) and (10) and take them as the first timestep's input of the decoder rather than a start token.

$$T_{sub} = \sum_{p=s}^{s+i} \varphi(w_p) \cdot Emb(w_p) \tag{9}$$

$$T_{obj} = \sum_{q=o}^{o+j} \varphi(w_q) \cdot Emb(w_q) \tag{10}$$

where the $\varphi(w_*)$ indicates the frequency of occurrence of the word w_*.

We employ a single-layer unidirectional GRU as the decoder. At each step t of the decoding stage, the decoder computes the current time step's attention distribution a^t based on the decoder state s_t and the encoder's output H. The attention distribution a^t is calculated with the method of Bahdanau, Cho, and Bengio (2015):

$$e_i^t = v^T \tanh\left(W_h h_i + W_s s_t + b_{\text{attn}}\right), \tag{11}$$

$$a^t = \text{softmax}\left(e^t\right), \tag{12}$$

where v, W_h, W_s and b_{attn} are learnable parameters.

And the attention distribution a^t is regarded as a relevance distribution between the decoder's input of the step t and the output of the encoder H. The attention distribution a^t is then utilized to compute a weighted sum of the encoder hidden states, known as the context vector c_t.

$$c_t = \sum_i a_i^t h_i \tag{13}$$

This context vector c_t will be concatenate with the decoder state s_t and fed into the decoder to compute the vocabulary distribution P_{vocab}.

$$P_{\text{vocab}}(w') = \text{softmax}\left(V'\left(V\left[s_t, c_t\right] + b\right) + b'\right) \tag{14}$$

where V, V', b and b' are learnable parameters. It is worth noting that the P_{vocab} represents a probability distribution over the vocabulary and indicates the final predicted word is w'.

Besides, we introduce the pointer mechanism (See, Liu, & Manning, 2017) to ensure the valuable contextual information such as the words "*president*" and "*city*" in Fig. 1, will not be discarded during generation stage. If the prompt generator generates some invalid special tokens, the pointer will copy a word from the original input based on the attention distribution a^t as the trigger word. The generation probability $p_{gen} \in [0, 1]$ will be obtained by the context vector c_t, the decoder state s_t and the decoder input x_t:

$$p_{\text{gen}} = \sigma\left(w_c^T c_t + w_s^T s_t + w_x^T x_t + b_{\text{pt}}\right) \tag{15}$$

where the w_c, w_s, w_x and scalar b_{pt} are learnable parameters and σ is the sigmoid function. The next step involves using p_{gen} as a soft gate to determine whether to generate a new word from the vocabulary by sampling from P_{vocab}, or to copy a word from the input sentence as shown in Eq. (16).

$$P_{\text{vocab}}(pw_t) = p_{\text{gen}}\ P_{\text{vocab}}(w') + \left(1 - p_{\text{gen}}\right) \sum_{i:w_i=w'} a^t, \tag{16}$$

where $P_{\text{vocab}}(w')$ denotes the probabilities distribution overall vocabulary outputted by decoder and the final predicted word is w'. Note that if the predicted trigger word w' is the special tokens of BERT (such as "[UNK]","[PAD]", etc.), then the $P(w')$ will be set as zero; similarly, if w' does not appear in the input sequence, then $\left(1 - p_{\text{gen}}\right) \sum_{i:w_i=w'} a^t$ will be set as zero.

Once we have generated all the trigger words $\{pw_1, pw_2, \ldots, pw_m\}$, we need to embed them into prompt tokens [Head_Prompt] and [Tail_Prompt] using context embedding:

$$[\text{Head\_Prompt}] = \sum_{i=1}^{u} \varphi(pw_i^{sub}) \cdot Emb(pw_i^{sub}) \tag{17}$$

$$[\text{Tail\_Prompt}] = \sum_{i=1}^{v} \varphi(pw_i^{obj}) \cdot Emb(pw_i^{obj}) \tag{18}$$

where the pw_i^{sub} and pw_i^{obj} denotes the i generated trigger word when the first timestep's input of the decoder is T_{sub} and T_{obj} respectively. And the u and v is the number of the trigger words.

3.3. In-domain adaptive pre-training

The existing works (Chen, Liu et al., 2022) have demonstrated that the performance of prompt-tuning decreases when the pre-trained corpus and the training datasets belong to different domains. In this section, we proposed an in-domain adaptive pre-training strategy to inject the domain-relevance knowledge by retrieving the related corpus from large external corpus(SST2 (Socher et al., 2013), DBPedia (Zhang, Zhao & LeCun, 2015), AGNews (Zhang, Zhao et al., 2015)) for improving the domain adaptation of the PLMs. Inspired by existing search-based pretraining methods (Wang et al., 2022), we found that the best matching algorithm BM25 score (Robertson, Zaragoza, et al., 2009; Schütze, Manning, & Raghavan, 2008) is suitable for retrieving the domain-related data from the corpus. Thus, we sample a small-scale training data from training data as the query Q and use BM25 to obtain the domain-relevance corpus from the large scale corpus d by Pyluence.[1]

$$BM25(Q, d) = \sum_i^n W_i R(q_i, d) \tag{19}$$

$$R(q_i, d) = \frac{f_i \cdot (k_1 + 1)}{f_i + k} \cdot \frac{qf_i \cdot (k_2 + 1)}{qf_i + k_2} \tag{20}$$

where W_i denotes the IDF weight of q_i; $k1, k2, b$ are adjustment factors, which are usually $k1 = 2, b = 0.75$; f_i indicates the frequency of q_i in d, and qf_i is the frequency of q_i in the input sentence.

After retrieving domain-specific text from the public corpus, we initiate incremental pretraining on PLMs. Since the training corpus consists of sentences, applying Roberta's original masking strategy for preprocessing would result in few words being masked within sentences. As a solution, we introduce the N-gram masking strategy. In particular, as illustrated in Fig. 4, we increase the probability of each word being selected from 15% to 30%. Next, we select a value N from the set 1, 2, 3. Assuming $N = 3$, the two words following the selected word had a 100% probability of being chosen for masking. Afterwards, the selected word had an 80% probability of being masked, a 10% probability of being randomly replaced, or replaced with a synonym. Furthermore, if the previous word was masked, the probability of the next word being masked was set to 0%. Through the aforementioned masking strategy, PLMs can more effectively acquire external knowledge from domain-relevant corpora.

3.4. Joint contrastive loss

To ensure the convergence, we introduce a Joint Contrastive Loss to optimize our model. After obtaining the prompt tokens from the context-aware pretrained prompt generator, we concatenate them with the original input to create x_{in} which will be fed into the PLMs for prompt-tuning.

$$prompt(x) = P_{sub}\ [\text{Head\_Prompt}]\ [\text{MASK}]\ [\text{Tail\_Prompt}]\ P_{obj} \tag{21}$$

$$x_{in} = [\text{CLS}]\ x\ [\text{SEP}]\ prompt(x)\ [\text{SEP}] \tag{22}$$

Next, we separately use the outputs of the last layer for [CLS] and [MASK] tokens to calculate the cross-entropy loss(CE Loss) $\mathcal{L}_{CE}$ and the contrastive loss(CT Loss) $\mathcal{L}_{CT}$. It is worth noting that when calculating the CE loss, we need to take the pre-extracted result α into account.

$$y_{cls} = \text{Softmax}(\mathcal{M}([\text{CLS}]|x_{in})) + \alpha \tag{23}$$

$$y_{mask} = \text{Softmax}(\mathcal{M}([\text{MASK}]|x_{in})) \tag{24}$$

$$\mathcal{L}_{CE} = \text{Cross-Entropy}(y_{cls}, y_i) \tag{25}$$

[1] https://lucene.apache.org/pylucene/.

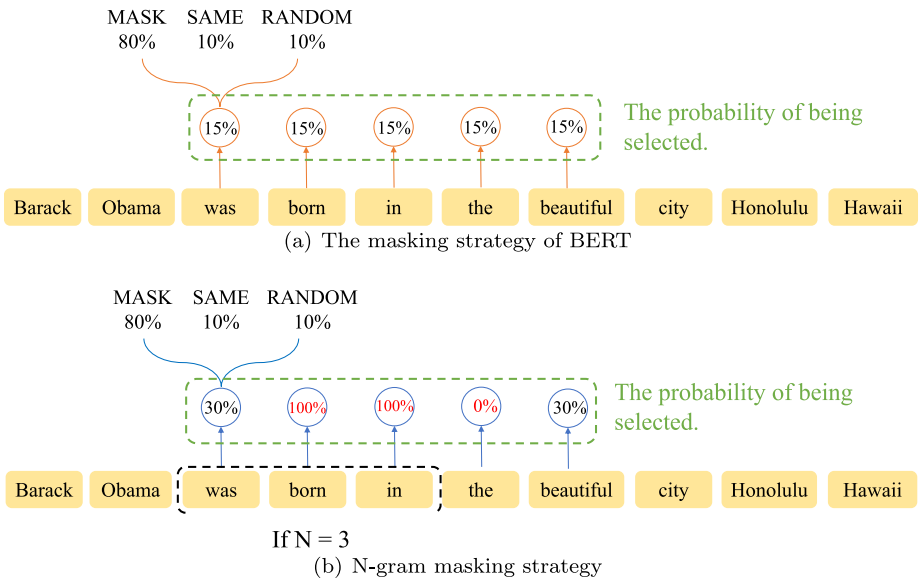


Fig. 4. Comparison of the Masking strategy. The N-gram Masking strategy is also used by GAP, which is better at learning phrase information as shown in the figure. "Mask" represents the token will be replaced with the special token [MASK], "Same" represents the token will remain unchanged, "RANDOM" represents the token will be replaced with random tokens in input sentence.

where y_i represents the label of the input.

On the other hand, we believe that the better the predicted value of the [MASK] token aligns with the semantics of the relation label, the better the relation extraction performance. For example, when we predict the word "birthday", the entities in the sentence are likely to have a "born" related relationship. If the predicted word is something like "blank", it becomes more challenging to determine the final relationship type. It is worth noting that the prediction for the [MASK] token is a word vector, not a probability distribution. Therefore, we cannot use a cross-entropy loss to minimize the distance between the predicted value and the word we expect to generate. Hence, we introduce the contrastive loss to make the semantics of the predicted value for the [MASK] token closer to the correct label's semantics and farther from the incorrect label's semantics, further enhancing the model's performance.

$$\mathcal{L}_{CT} = \frac{1}{N} \ln \frac{\exp(s(y_{mask}, y^+)/\tau)}{\exp(s(y_{mask}, y^+)/\tau) + \sum \exp(s(y_{mask}, y^-)/\tau)} \tag{26}$$

where y^+ represents the embedding corresponding to the correct relation label, and y^- represents the embedding corresponding to the incorrect relation label.

Finally, we use Kullback-Leibler (KL) divergence as a balancing factor between the two loss functions to obtain the ultimate Joint Contrastive Loss. In Eq. (27), the KL divergence represents the consistency between the pre-extraction and PLMs' extraction results. If the two extraction results are inconsistent, the current sample is a challenging sample. The KL divergence is bigger than 0 at that time, and the contrastive loss can benefit more from this challenging sample.

$$\mathcal{L} = \mathcal{L}_{CE} + (1 + \mathrm{KL}(y_i, y_{cls})) \cdot \mathcal{L}_{CT} \tag{27}$$

4. Experiments

4.1. Datasets and evaluation metrics

In this section, we conduct some experiments on four RE datasets to show the effectiveness of our model: SemEval 2010 Task 8 (SemEval) (Hendrickx et al., 2019), TACRED (Zhang, Zhong, Chen, Angeli, & Manning, 2017), TACRED-Revisit (Alt, Gabryszak, & Hennig, 2020), Re-TACRED (Stoica, Platanios, & Póczos, 2021). Statistical details are provided in Table 1.

Table 1
Statistics for the public RE datasets used for our experiment, including numbers of relations and instances in the different split.

Datasets	Train	dev	Test	rel
SemEval	6507	1493	2717	19
TACRED	68 124	22 631	15 509	42
TACREV	68 124	22 631	15 509	42
ReTACRED	58 465	19 584	13 418	40

Evaluation Metrics We adopt the official evaluation metric of these four datasets, which is based on the macro-averaged F1 score (excluding Other relation). Furthermore, the reason we do not use accuracy as an evaluation metric in relation extraction tasks is because it is highly susceptible to the impact of imbalanced data samples. This means that a model achieving a high accuracy on a dataset does not necessarily confirm its ability to correctly identify and classify relation types. Compared to accuracy, precision and recall provide a more powerful indication of the model's classification ability. Precision represents how many of the identified relationships are correct, while recall represents how many of all the samples with relationships have been identified. And F1 score is computed as the harmonic mean of accuracy and recall, which provides a comprehensive picture of performance of the model.

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN} \tag{28}$$

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{29}$$

where TP (True Positives) represents the number of instances where the model correctly predict the positive class, FP (False Positives) represents the number of instances where the model incorrectly predict the positive class when the actual class is negative and FN (False Negatives) represents the number of instances where the model incorrectly predict the negative class when the actual class is positive.

4.2. Implementation details

We present the details of the model implementation in this subsection, which is based on the Transformers and Huggingface package. we use Roberta-large as the base PLMs, and choose AdamW as the

Table 2
The F1 scores (%) of RE on different datasets in supervised setting. The best results are bold.

Model	Extra data	SemEval	TACRED	TACREV	ReTACRED
Fine-tuning-[Roberta]	w/o	87.6	68.7	76.0	84.9
R-BERT (Wu & He, 2019)	w/o	89.3	69.4	–	–
SpanBERT (Joshi et al., 2020)	w/	–	70.8	78.0	85.3
KnowBERT (Peters et al., 2019)	w/	89.1	71.5	79.3	89.1
LUKE (Yamada et al., 2020)	w/	–	72.7	80.6	–
MTB (Soares et al., 2019)	w/	89.5	70.1	–	–
GDPNet (Xue, Sun, Zhang, & Chng, 2021)	w/o	–	71.5	79.3	–
RE-DPM (Tian, Song, & Xia, 2022)	w/o	89.9	71.5	79.3	–
SPOT (Li et al., 2022)	w/o	89.4	–	–	89.4
RELA (Li et al., 2023)	w/o	89.6	71.2	79.7	–
PTR-[Roberta] (Han et al., 2022)	w/o	89.9	72.4	81.4	90.9
KnowPrompt (Chen, Zhang et al., 2022)	w/o	90.1	72.4	81.7	91.1
GAP (ours)	**w/**	**90.3**	**72.7**	**82.7**	**91.4**

optimizer for this model. We set the learning rate to 1e−4 for the SemEval dataset and 5e−5 for the other datasets, with a linear warmup for the first 10% of the training steps. Additionally, we conduct a series of experiments to find the optimal τ. All experiments in this paper are performed on dual RTX 3090 GPUs and verified several times to obtain the best set of hyperparameters.

4.3. Pretraining setting

In terms of pretraining, we pretrain the prompt generator with the CNN Dailymail Dataset (Hermann et al., 2015), which is widely used in text summarization tasks. The CNN Dailymail dataset contains 287,113 training samples, 13,368 validation samples, and 11,490 test samples. The evaluation metrics used for the text summarization task are ROUGE-1, ROUGE-2, and ROUGE-L. In the pre-training phase, our prompt generator achieved a 38.76 ROUGE-1 value, a 15.60 ROUGE-2 value, and a 35.81 ROUGE-L value, demonstrating that our generator has the ability to extract and generalize the semantics of sentences.

4.4. Baselines

According to the training paradigm, the existing studies can be roughly divided into the following two categories:

(1) fine-tuning paradigm has achieved promising results on RE by using the annotated datasets to train the well-designed RE classifier.

(2) The knowledge-enhanced fine-tuning paradigm not only consider how to train the classifier but also inject the entity information or external knowledge into the model, such as SpanBERT (Joshi et al., 2020), LUKE (Yamada et al., 2020), and RELA (Li, Yu, Ye, Zhang, & Zhang, 2023), among others.

(3) The prompt-tuning paradigm focus on extracting the relations by prompting information to stimulate the latent knowledge of PLMs, such as PTR (Han et al., 2022), KnowPrompt (Chen, Zhang et al., 2022).

Thus, we make a comparative analysis between PTR with our method in supervised and few-shot setting.

4.5. Comparison experiments

To validate the superiority of our method, we conduct experiments on the four datasets mentioned earlier. We perform comparative experiments in both supervised learning and few-shot learning settings, and the experimental results are shown in Tables 2 and 3.

4.5.1. Comparison with the SOTA methods in supervised setting

Table 2 presents our experimental results in supervised learning compared to the baseline. From the experiments in supervised setting, we can draw the following conclusions:

(1) Prompt-tuning-based methods easily achieve or even surpass state-of-the-art (SOTA) methods based on fine-tuning in the supervised setting. As shown in Table 2, the prompt-based relation extraction methods like PTR (Han et al., 2022), KnowPrompt (Chen, Zhang et al., 2022), and our proposed GAP outperformed fine-tuning-based methods in terms of F1 scores. These fine-tuning-based methods like GDPNet (Xue et al., 2021) and KnowBERT (Peters et al., 2019) have carefully designed downstream classifiers and introduced external knowledge to enhance model performance. However, their performance still lags behind prompt-tuning methods that do not use any additional knowledge, such as PTR, not to mention KnowPrompt and our GAP. The primary reason for this difference is the gap between the objectives of upstream and downstream tasks in fine-tuning paradigm, which hinders the downstream classifier from benefiting more from pre-training. This is where prompt-based methods have an advantage.

(2) Enhancing prompts with external knowledge effectively improves model performance in relation extraction. It is well-known that PTR does not introduce additional knowledge but relies on manually designed sub-prompts and logical rules for relation extraction. In contrast, KnowPrompt uses embeddings from the text in samples to enhance prompt tokens, which resulting in superior performance on multiple datasets compared to PTR. Particularly, KnowPrompt achieves a 1.3% and 1.4% F1 score improvement over PTR on the SemEval and TACRED datasets, respectively. However, KnowPrompt's entity distribution has some randomness within each mini-batch, leading to decreased performance when the data is shuffled less. On the other hand, our GAP enhances prompt tokens with context information or trigger words extracted or generated from sentences, effectively injecting prior knowledge while maintaining stable model performance.

(3) Our proposed GAP effectively reduces manual or computational costs in template engineering and achieved SOTA results in supervised setting. Unlike PTR, we do not require domain experts to design prompt templates for datasets. Instead, we generate them automatically using the prompt generator. Compared to AutoPrompt, we do not need extensive computational resources and time because our prompt generator is lightweight, and we have pre-trained it using publicly available corpora.

4.5.2. Comparison with the SOTA methods in few-shot setting

We also conduct experiments in few-shot setting, and the experimental results are presented in Table 3.

We can draw similar conclusions to those in Table 2 under the supervised setting. Additionally, we have noted the following:

(1) Our method exhibits a more pronounced advantage in the few-shot setting. From Table 3, we can observe that our method achieves F1 scores that are 16.6%, 13.2%, 11.1%, and 15.3% higher than the fine-tuning method on the four datasets, respectively. We attribute this improvement to prompt-tuning, as PTR also demonstrates similar experimental results. The main reason behind this is that prompt-tuning effectively leverages the knowledge learned by PLMs during the pre-training phase to tackle downstream tasks. Consequently, it can deliver substantial performance even in low-resource scenarios.

Table 3
The F1 scores (%) of RE on different test sets in supervised setting. We set K = 8, K = 16 and K = 32 for few-shot experiment. The subscript of "↓" means that GAP did not over the best results of baselines and the "↑" means it over the best result of baselines. Best results are bold.

Dataset	Method	K = 8	K = 16	K = 32	Mean
Semeval	Fine-tuning	41.3	65.2	80.1	62.2
	GDPNet (Xue et al., 2021)	42	67.5	81.2	63.6
	AdaPrompt (Chen, Liu et al., 2022)	–	–	–	–
	PTR (Han et al., 2022)	**70.5**	**81.3**	84.2	**78.4**
	GAP (ours)	70.2↓	80.9↓	**85.3**↑	78.8↑
TACRED	Fine-tuning	12.2	21.5	28	20.6
	GDPNet (Xue et al., 2021)	11.8	22.5	28.8	21.1
	AdaPrompt (Chen, Liu et al., 2022)	–	–	–	–
	PTR (Han et al., 2022)	28.1	30.7	32.1	30.3
	GAP (ours)	**31.9**↑	**34.5**↑	**35.0**↑	**33.8**↑
TACREV	Fine-tuning	13.5	22.3	28.2	21.4
	GDPNet (Xue et al., 2021)	12.3	23.8	29.1	21.8
	AdaPrompt (Chen, Liu et al., 2022)	25.2	27.3	30.8	27.8
	PTR (Han et al., 2022)	28.7	31.4	32.4	30.8
	GAP (ours)	**30.7**↑	**34.1**↑	**35.7**↑	**33.5**↑
Re-TACRED	Fine-tuning	28.5	49.5	56	44.7
	GDPNet (Xue et al., 2021)	29.0	50.0	56.5	45.2
	AdaPrompt (Chen, Liu et al., 2022)	–	–	–	–
	PTR (Han et al., 2022)	51.5	56.2	62.1	56.6
	GAP (ours)	**53.9**↑	**60.6**↑	**65.5**↑	**60.0**↑

Table 4
The input example of the marker-based methods.

Methods	Input samples
Entity marker	President [E1] Barack Obama [/E1] was born in the beautiful city of [E2] Honolulu, Hawaii [/E2].
Typed entity marker	President ⟨S:PERSON⟩ Barack Obama ⟨/S:PERSON⟩ was born in the beautiful city of ⟨O:CITY⟩ Honolulu, Hawaii ⟨/O:CITY⟩.
Entity marker (punct)	President @ Barack Obama @ was born in the beautiful city of # Honolulu, Hawaii #.
Typed entity marker (punct)	President @ * person * Barack Obama @ was born in the beautiful city of # ^ city ^ Honolulu, Hawaii #.
GAP (ours)	President Barack Obama was born in the beautiful city of Honolulu, Hawaii. Barack Obama [Head_Prompt] ⟨$MASK$⟩ [Tail_Prompt] Honolulu, Hawaii.

(2) We observe that our method's performance on the SemEval dataset under the K = 8 and K = 16 settings was not satisfactory. Therefore, we conduct an error analysis on the SemEval dataset and identified two main reasons: On one hand, the relationship labels in SemEval require more in-depth reasoning for accurate classification. In low-resource scenarios, our model struggles to accurately classify complex relationship types because we still require a certain amount of data to train the prompt generator. On the other hand, the SemEval dataset contains a significant amount of mislabeling and noise. For example, here is one of the samples from the training set, with the entity pairs *(residence, factory)* and relation labels *"Component-Whole(e2,e1)"*:

"The factory's workshop functioned inside an extension which was bigger than the actual residence".

As we can see, the entity *"factory"* and *"residence"* are without any relation in the sentence, but their label is *"Component-Whole(e2,e1)"* which is the relation between the *"factory"* and the *"workshop"*. Furthermore, we can observe that the surrounding context of the sentences does not provide additional valuable information about the entities. This has led to prediction errors in our model. However, overall, our model still exhibits a significant advantage in the few-shot setting.

4.5.3. Comparison with the marker-based methods

The prompt tokens used in GAP have some similarities with Typer Marker and Entity Marker (Zhou & Chen, 2021). Typer Marker and Entity Marker (Zhou & Chen, 2021) have been widely employed in existing relation extraction tasks, primarily to emphasize the positional information of entities and make the model focus more on them. The input examples of the marker-based method can be seen in Table 4. On the other hand, GAP embeds effective trigger words from the context into its prompt tokens for prompt-tuning. To compare the performance differences between the two approaches, we conduct comparative experiments.

Table 5
The F1 scores (%) of RE on different marker-based models in supervised setting. "w/o" means that no additional data is used for pretraining and fine-tuning, yet "w/" means that the model uses extra data for tasks. The subscript of "↓" means that APP did not over the best results of baselines and the "↑" means it over the best result of baselines.

Model	Extra data	SemEval	TACRED	TACREV	ReTACRED
Entity marker	w/o	–	70.7	80.2	90.5
Type marker	w/	–	71.0	80.8	90.5
Entity marker (punct)	w/o	–	71.4	81.2	90.5
Type marker (punct)	w/	–	74.6	83.2	91.1
PTR-[Roberta] (Han et al., 2022)	w/o	89.9	72.4	81.4	91.1
GAP (ours)	**w/**	**90.3**↑	**72.7**↑	**82.7**↑	**91.4**↑

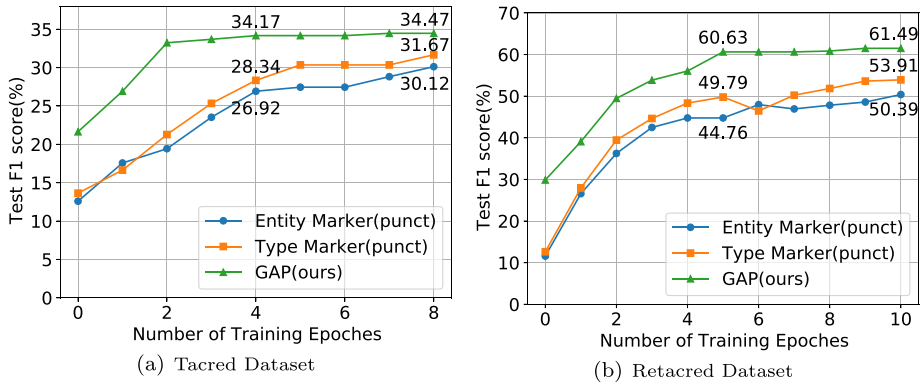
From Table 5, we can observe that our model maintains its advantage compared to marker-based relation extraction methods. This is as expected because the prompt tokens generated by GAP are more flexible compared to fixed markers. Additionally, we enhance these tokens with context, allowing them to perform better in prompt-tuning. As shown in Figs. 5(a) and 5(b), the experimental results in the few-shot setting also confirm the same conclusion.

4.6. Ablation study

4.6.1. Effect of the proposed modules

To validate the effectiveness of the Context-aware Prompt Generator (CPG), we will not use the context-enhanced prompts but instead use the manual prompts shown in Fig. 1. To validate the effectiveness of the Joint Contrastive Loss (JCL), we will replace it with a cross-entropy loss function. The ablation experimental results are shown in Table 6.

Furthermore, from Table 6, we can observe that the majority of GAP's performance improvement comes from the Context-aware



(a) Tacred Dataset (b) Retacred Dataset

Fig. 5. Experimental results of different marker-based methods in the few-shot setting of K = 16.

Table 6
The F1 scores (%) of the ablation study in supervised setting. Legend: **CPG**: Context-aware Prompt Generator; **IAP**: In-domain Adaptive Pretraining; **JCL**: Joint Contrastive Loss.

Module			Datasets	
CPG	IAP	JCL	SemEval	TACREV
			88.7	81.5
√			89.7 (+1.0)	82.3 (+0.8)
	√		88.7 (+0.0)	81.6 (+0.1)
		√	89.1 (+0.4)	81.9 (+0.4)
√	√		89.9 (+1.2)	82.5 (+1.0)
√		√	90.1 (+1.4)	82.5 (+1.0)
	√	√	89.5 (+0.8)	82.1 (+0.6)
√	√	√	90.3 (+1.6)	82.7 (+1.2)

Prompt Generator (CPG), which provides enhanced prompt tokens with prior knowledge to boost GAP's performance. On the other hand, we find that in our experiments, the In-domain Adaptive Pretraining module (IAP) brings only a small improvement to the model. It results in only a 0.0% F1 score improvement in the SemEval dataset and a 0.1% improvement in the TACREV dataset. The primary reason for this is that the size of the public corpus we used may not be large enough, and it might not contain domain-specific knowledge. Therefore, we believe that using domain-specific corpora for in-domain pretraining could lead to more significant improvements with this module.

4.6.2. Effect of the temperature parameter

The temperature parameter's role is to adjust the model's focus on challenging samples. A smaller temperature parameter in the joint contrastive loss makes the model pay more attention to challenging negative samples. In our ablation experiments on τ, initially, we set τ to values between 0.1 and 0.4, as shown in Fig. 6(a). We can see that as τ increases, the performance of the experiment results decreases, with the best performance achieved at the minimum value of 0.1. Therefore, we set the range of τ to $[0.08, 0.12]$ to explore the optimal τ, as shown in Fig. 6(b). We find that when $\tau = 0.09$, three datasets achieved local optimal performance, while when $\tau = 0.10$, two datasets achieved the best performance.

In summary, when GAP focuses more on complex negative samples, it can achieve better performance. However, if the temperature parameter τ is set too small, the model will focus more on particularly challenging negative samples, which may actually be potential positive samples. This will make the model hard to converge or result in poor generalization.

4.6.3. Effect of the pretrained language models

For prompt-tuning, the choice of pre-trained language models (PLMs) can directly impact the model's effectiveness. This is because, compared to fine-tuning, prompt-tuning relies more on the knowledge learned by PLMs during the pre-training phase. To investigate the impact of different PLMs on GAP, we conduct experiments without in-domain pretraining and used various PLMs. As shown in Fig. 7, in different experimental settings and datasets, Roberta outperforms BERT (Devlin et al., 2018) and GPT-2 (Radford et al., 2019) consistently. On one hand, as an improved version of BERT (Devlin et al., 2018), it is expected that Roberta (Liu et al., 2019) performs better in these experiments. Research has found that Roberta's decision to forgo the Next Sentence Prediction (NSP) task during pretraining actually benefits its performance in pretraining, which is one of the reasons why Roberta (Liu et al., 2019) performs better in few-shot settings. Since both BERT (Devlin et al., 2018) and Roberta (Liu et al., 2019) employ the same bidirectional Transformer architecture, the performance gap between them narrows as the amount of training data increases. On the other hand, the lower performance of GPT-2 (Radford et al., 2019) may be attributed to the fact that GPT-2 (Radford et al., 2019) is a unidirectional language model. GAP employs symmetric prompts, and as a result, when GPT-2 predicts the [MASK] token, it only considers the head prompt while ignoring the tail prompt. This leads to its performance not being comparable to Roberta (Liu et al., 2019).

4.7. Qualitative analysis

To qualitatively demonstrate the effectiveness of GAP, we conduct qualitative experiment on GAP and two other baselines.

Compared to existing methods, the greatest advantage of our approach is its ability to provide more accurate and effective contextual prior knowledge. This capability guides the PLMs to predict relationships between entities more accurately. To effectively demonstrate this advantage, we conduct a quality analysis as shown in Table 7. In the first example, due to the presence of a relational pronoun "she", PTR incorrectly predicted the relationship between "she" and "Honolulu" as "per:identity". This may be because PTR primarily relies on entity type information and logical rules, overlooking the importance of context. While KnowPrompt enhances prompt tokens through entity embeddings, it also fails to focus on the details of the context, leading to prediction errors. For instance, in the fifth example, KnowPrompt directly classified "founder" as "org:founded_by" simply because it appeared, a similar situation occurring in the second and sixth examples. On the other hand, in samples with shorter contexts, KnowPrompt is also more prone to errors, as demonstrated in the third and fourth examples. As shown in Table 7, our proposed method, GAP, accurately predicts relationships in the mentioned examples. It introduces entity-related trigger words rather than solely considering entity types and embeddings. This enables the model to pay attention to contextual details and make more precise predictions.

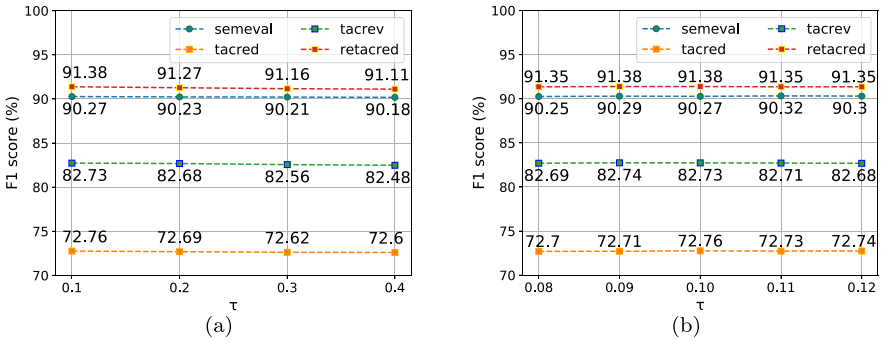


Fig. 6. The comparison of different value to the parameter τ.

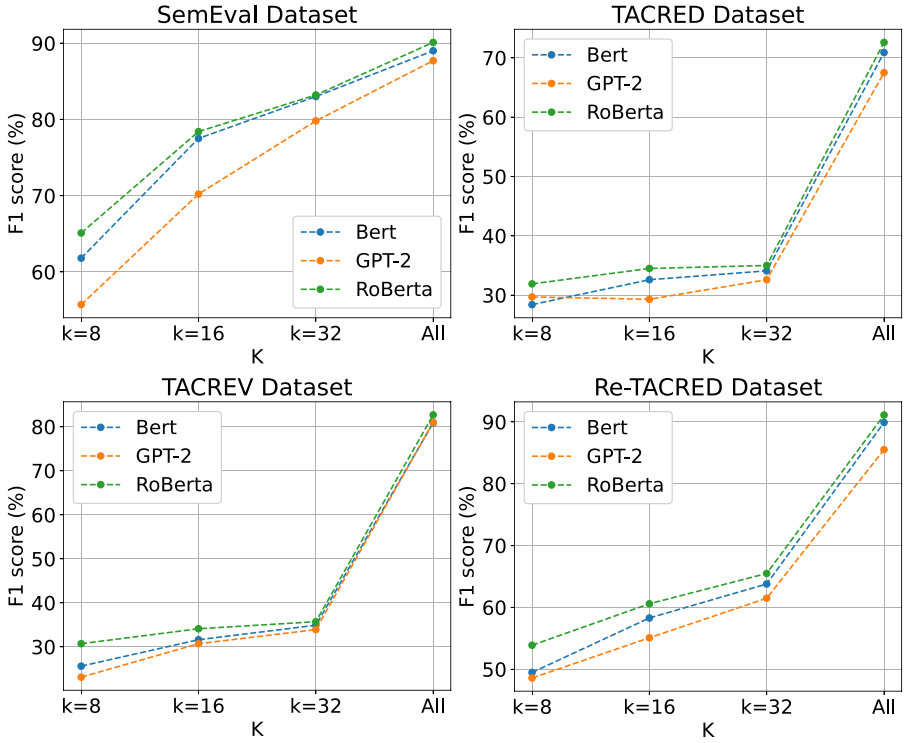


Fig. 7. Performance comparison of GAP using different pre-trained language models for relation extraction.

4.8. Visualization analysis

To validate whether our pre-extraction module can pre-extraction the relations by using context information, we use the sentence *"President Barack Obama was born in the beautiful city of Honolulu, Hawaii"*. as input and visualized the *label-to-context relative metric* M_{l2w} before softmax. The visualization results are shown in Fig. 8. In the sentence, the words *"president"*, *"city"* and *"born"* are important context or trigger words that can determine the relationship between "Barack Obama" and "Honolulu, Hawaii". In Fig. 8, these three words also show strong correlations with their corresponding labels, effectively demonstrating the effectiveness of our pre-extraction module.

To provide a more intuitive illustration of the effectiveness of context-aware prompt tokens generated by GAP, we randomly select a batch of test data and conduct a visualization experiment using t-SNE. As shown in Fig. 9, the blue triangles in the graph represent the predictions of [MASK] without using context-aware prompt tokens generated by GAP, while the red "+" symbols represent the experimental results when these prompt tokens are used. The yellow squares represent the embeddings of relation labels. We can observe that the predictions with context-aware prompt tokens (red "+") are closer to their corresponding relation labels (yellow squares), indicating that GAP-generated context-aware prompt tokens guide PLMs to make predictions that are more suitable for the current task.

5. Discussion

5.1. Variants of our method

GAP generates trigger words by mapping the output vector of the decoder through a linear layer and a softmax function to create a probability distribution over the vocabulary. We have considered whether GAP might achieve better results by generating continuous prompt vectors instead of trigger words. Therefore, we introduce a new linear layer to directly map the decoder's output vector to a semantic space to obtain prompt vectors. We also use the same context embedding operation to create new prompt tokens. Following this approach, we introduce a variant of GAP, which we refer to as "GAP-[Continuous]", while the original GAP is referred to as "GAP-[Discrete]".

As shown in Fig. 10 and Table 8, GAP-[Discrete] outperformed GAP-[Continuous] in both experimental settings. We believe that the

Table 7
Qualitative analysis results of our GAP and two baselines. Entities that need to be extracted their relations are marked in red.

Sentence:	Growing up on Oahu in the 1970s, she hung out at the Scientology community church. Or org in downtown Honolulu.
Ground True:	no_relation
PTR:	per:identity ×
KnowPrompt:	per:city_of_birth ×
GAP (ours):	no_relation ✓
Sentence:	Icey Simmons, 85, a resident of the Silver Crest Nursing Center, said she would see Daniels nearly every weekend when Simmons would go to the church for Sunday service or when Daniels would drop by.
Ground True:	no_relation
PTR:	no relation ✓
KnowPrompt:	per:date_of_birth ×
GAP (ours):	no_relation ✓
Sentence:	The grip is fitted over a rear part of a core of the helve of the hammer.
Ground True:	Component-Whole(e1,e2)
PTR:	Content-Container(e2,e1) ×
KnowPrompt:	Component-Whole(e1,e2) ✓
GAP (ours):	Component-Whole(e1,e2) ✓
Sentence:	The bottles have leaked poison into the milk.
Ground True:	Entity-Destination(e2,e1)
PTR:	no relation ×
KnowPrompt:	Entity-Origin(e2,e1) ×
GAP (ours):	Entity-Destination(e2,e1) ✓
Sentence:	Donald Wildmon, the founder and head of the American Family Association, is asking its members to petition Congress to end all funding for PBS.
Ground True:	per:employee_of
PTR:	org:founded ×
KnowPrompt:	org:founded_by ×
GAP (ours):	per:members ✓
Sentence:	President Lee Teng-hui confers the Order of the Brilliant Star with a Violet Grand Cordon on Samuel Noordhoff, founder of the Noordhoff Craniofacial Foundation.
Ground True:	org:founded_by
PTR:	org:founded ×
KnowPrompt:	org:founded_by ✓
GAP (ours):	org:founded_by ✓

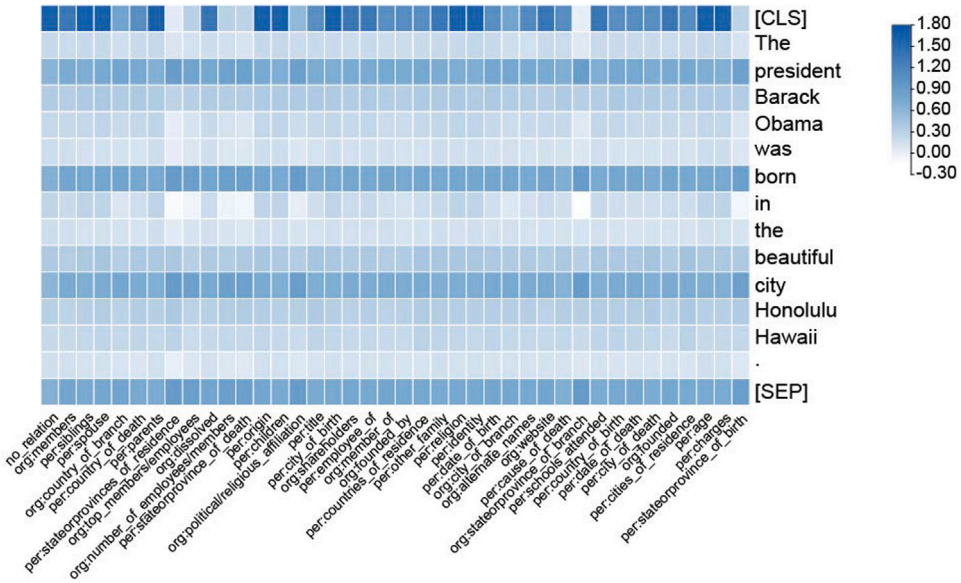


Fig. 8. The heatmap of the label-to-context relative metric M_{l2w} before softmax.

primary reason for this experimental result is that our prompt generator was pretrained by a text summarization task. Therefore, directly changing the model structure of the prompt generator will disrupt its contextual understanding capabilities, leading to a decrease in model performance.

5.2. The selection of the structure of the prompt generator

In the early stages of our research, we explored which neural network architecture would be suitable for generating prompts. We attempted to use the Transformer (Vaswani et al., 2017) as the prompt generator, but we found that the prompt tokens it generated did not improve the performance of relation extraction tasks. We observed that a single-layer Transformer (Vaswani et al., 2017) had relatively weak non-linear capabilities as it primarily used multi-head attention mechanism to weight the original input. Furthermore, it has been pointed out in existing literature (Blin & Kucharavy, 2021; Buestán-Andrade, Santos, Sierra-García, & Pazmiño-Piedra, 2023) that despite the use of positional encoding, Transformer still exhibits differences in sequence modeling capabilities compared to recurrent structures like

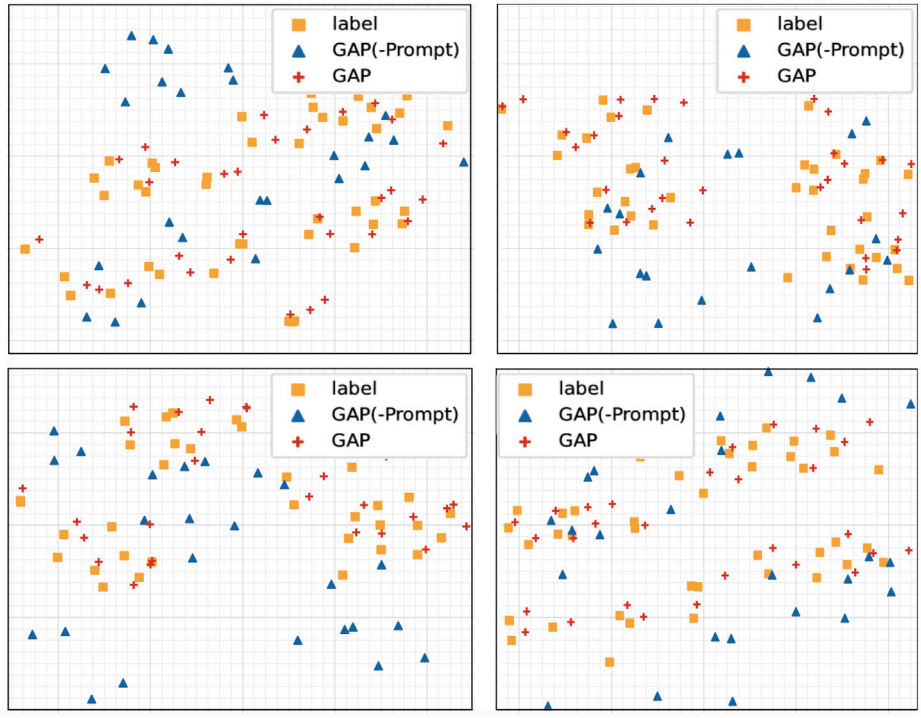


Fig. 9. Visualization of the typical samples, persudo label embedding and predicted [MASK] token by t-SNE. The blue triangles represent the results of relation extraction without using the prompt tokens generated by GAP, whereas the red plus signs (+) indicate the results using our generated prompt tokens. It is evident that using our prompts leads to embeddings of predicted [MASK] that are closer to the actual relation labels.

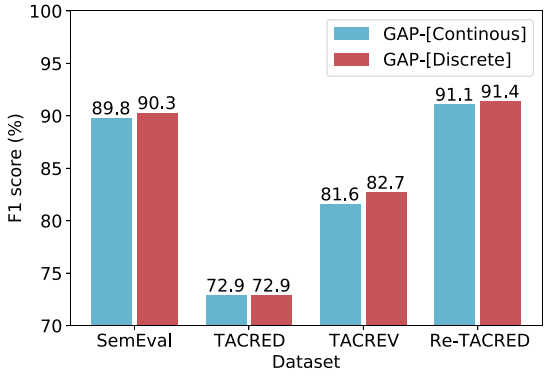


Fig. 10. The comparison between GAP-[Discrete] and GAP-[Continuous] for RE in supervised settings.

Table 8
The comparison between GAP-[Discrete] and GAP-[Continuous] for RE in few-shot settings. The better results are bold.

Dataset	Type	K = 8	K = 16	K = 32	Mean
SemEval	GAP-[Discrete]	**70.2**	**80.9**	**85.3**	**78.8**
	GAP-[Continuous]	69.8	81.3	83.9	78.4
TACRED	GAP-[Discrete]	**31.9**	**34.5**	**35.2**	**33.8**
	GAP-[Continuous]	31.7	34.0	34.7	33.5
TACREV	GAP-[Discrete]	**30.7**	**34.1**	**35.7**	**33.5**
	GAP-[Continuous]	29.8	34.0	34.8	32.9
Re-TACRED	GAP-[Discrete]	**53.9**	**60.6**	**65.5**	**60.0**
	GAP-[Continuous]	53.4	59.1	64.5	59.0

RNNs. Moreover, in contrast to large pre-trained Transformers, a single-layer Transformer does not exhibit an advantage in some tasks (Li, Wallace et al., 2020; Siino, Di Nuovo, Tinnirello, & La Cascia, 2022). However, multi-layer Transformers increased network depth, making it challenging for the model to converge. Additionally, multi-layer Transformers introduce more parameters, which are not friendly to relation extraction in low-data settings.

We also tried Prefix-tuning (Li & Liang, 2021) and P-tuning (Liu, Zheng et al., 2021) for relation extraction, but since they freeze the parameters of PLMs, which limit the performance of relation extraction in supervised setting. Therefore, we consider them more suitable for fine-tuning Large Language Models (LLMs). While GRU can generate context related to the current text from the vocabulary as trigger words, a single-layer Transformer is limited to weighting the input text, allowing the model to focus on the parts that are more relevant to the relation category. In summary, we believe that GRU is a more suitable network architecture for serving as the prompt generator.

5.3. Advantages and limitations

Through the aforementioned experiments, our method exhibits several advantages:

(1) GAP eliminates the need for domain experts to design templates and can automatically generate context-enhanced prompt tokens through the prompt generator, introducing contextual prior knowledge.

(2) GAP does not suffer from the problem of incompatibility with the traditional LM objective. It enhances prompt tokens with trigger words that not only come from the input sentence but can also be retrieved from the vocabulary to augment prompt tokens. This makes our approach more aligned with upstream tasks compared to existing methods which insert the same prompt template for each sample.

(3) GAP demonstrates stronger domain adaptability. By using the in-domain adaptive pretraining strategy, it can easily be applied to relation extraction tasks in specific domains, such as biomedical relation extraction.

(4) GAP can use a joint contrastive loss function to boost the weighting of the challenging samples, enabling our model to better learn from these challenging instances and improve prediction accuracy.

However, our model still has its limitations. While the prompt generator can generate context-aware prompt tokens effectively, it also

increases the model's size, which diminishes its advantage in few-shot and zero-shot settings. Additionally, when we are unable to obtain domain-specific corpora or when the scale of domain-specific corpora is very small, the improvements brought by in-domain pretraining are not significant.

6. Conclusions

To overcome the limitations of traditional RE methods based on fine-tuning paradigm, we propose a novel prompt-tuning based RE method (GAP). Our proposed GAP has a prompt generator which generates more effective prompt tokens from entity and context information. These context-aware prompt tokens can better provide a priori knowledge to pretrained language models, improving the relation extraction capability. Besides, we propose the in-domain pretraining strategy to inject external knowledge for improving the robustness of our model. Furthermore, the joint contrastive loss function is to ensure our model can be better converged. According to the experimental results on four public datasets, our method has more advantages in both supervised and few-shot setting.

In our future work, we will explore strategies for effectively reducing the parameters of GAP and attempt to adapt it to other information extraction tasks. Additionally, we will explore how to automatically generate suitable prompt templates using Large Language Models (LLMs) and then further fine-tune the LLM with the generated prompts to transform it into a better information extractor.

CRediT authorship contribution statement

Zhenbin Chen: Conceptualization, Methodology, Investigation, Writing – original draft. **Zhixin Li:** Data curation, Methodology, Writing – original draft. **Yufei Zeng:** Review & editing. **Canlong Zhang:** Review & editing. **Huifang Ma:** Review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

This work is supported by National Natural Science Foundation of China (Nos. 62276073, 61966004, 62266009), Guangxi Natural Science Foundation, China (No. 2019GXNSFDA245018), Innovation Project of Guangxi Graduate Education, China (No. YCBZ2023055), Guangxi "Bagui Scholar" Teams for Innovation and Research Project, and Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

References

Alt, C., Gabryszak, A., & Hennig, L. (2020). TACRED revisited: A thorough evaluation of the TACRED relation extraction task. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 1558–1569).

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *3rd international conference on learning representations*.

Blin, K., & Kucharavy, A. (2021). Can the transformer be used as a drop-in replacement for RNNs in text-generating GANs? arXiv preprint arXiv:2108.12275.

Buestán-Andrade, P.-A., Santos, M., Sierra-García, J.-E., & Pazmiño-Piedra, J.-P. (2023). Comparison of LSTM, GRU and transformer neural network architecture for prediction of wind turbine variables. In *International conference on soft computing models in industrial and environmental applications* (pp. 334–343). Springer.

Bunescu, R., & Mooney, R. (2005). A shortest path dependency kernel for relation extraction. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing* (pp. 724–731). The Association for Computational Linguistics.

Cabot, P.-L. H., & Navigli, R. (2021). REBEL: Relation extraction by end-to-end language generation. In *Findings of the association for computational linguistics: EMNLP 2021* (pp. 2370–2381).

Chen, Y., Liu, Y., Dong, L., Wang, S., Zhu, C., Zeng, M., et al. (2022). AdaPrompt: Adaptive model training for prompt-based NLP. arXiv preprint arXiv:2202.04824.

Chen, K., & Sun, S. (2023). CP-rec: Contextual prompting for conversational recommender systems. *Vol. 37*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 12635–12643).

Chen, X., Zhang, N., Xie, X., Deng, S., Yao, Y., Tan, C., et al. (2022). Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM web conference 2022* (pp. 2778–2788).

Chia, Y. K., Bing, L., Poria, S., & Si, L. (2022). RelationPrompt: Leveraging prompts to generate synthetic data for zero-shot relation triplet extraction. In *Findings of the association for computational linguistics* (pp. 45–57).

Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., & Hu, G. (2017). Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th annual meeting of the association for computational linguistics* (pp. 593–602).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics* (pp. 4171–4186).

Dong, S., Zhan, J., Hu, W., Mohajer, A., Bavaghar, M., & Mirzaei, A. (2023). Energy-efficient hierarchical resource allocation in uplink-downlink decoupled NOMA HetNets. *IEEE Transactions on Network and Service Management*.

Gu, Y., Han, X., Liu, Z., & Huang, M. (2021). Ppt: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th annual meeting of the association for computational linguistics* (pp. 8410–8423).

Han, X., Zhao, W., Ding, N., Liu, Z., & Sun, M. (2022). Ptr: Prompt tuning with rules for text classification. *AI Open*, *3*, 182–192.

He, Z., Chen, W., Li, Z., Zhang, M., Zhang, W., & Zhang, M. (2018). See: Syntax-aware entity embedding for neural relation extraction. *Vol. 32*, In *Proceedings of the AAAI conference on artificial intelligence*.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Séaghdha, D. O., Padó, S., et al. (2019). Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. arXiv preprint arXiv:1911.10422.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., et al. (2015). Teaching machines to read and comprehend. *Advances in Neural Information Processing Systems*, *28*.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., & Levy, O. (2020). Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, *8*, 64–77.

Kambhatla, N. (2004). Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction. In *Proceedings of the 42nd annual meeting of the association for computational linguistics* (pp. 178–181).

Katiyar, A., & Cardie, C. (2016). Investigating LSTMs for joint extraction of opinion entities and relations. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*.

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Lee, J., Seo, S., & Choi, Y. S. (2019). Semantic relation classification via bidirectional lstm networks with entity-aware attention using latent entity typing. *Symmetry*, *11*(6), 785.

Li, C., Gao, F., Bu, J., Xu, L., Chen, X., Gu, Y., et al. (2021). Sentiprompt: Sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis. arXiv preprint arXiv:2109.08306.

Li, J., Katsis, Y., Baldwin, T., Kim, H.-C., Bartko, A., McAuley, J., et al. (2022). SPOT: Knowledge-enhanced language representations for information extraction. In *Proceedings of the 31st ACM international conference on information & knowledge management* (pp. 1124–1134).

Li, X. L., & Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing* (pp. 4582–4597).

Li, Z., Peng, Z., Tang, S., Zhang, C., & Ma, H. (2020). Text summarization method based on double attention pointer network. *IEEE Access*, *8*, 11279–11288.

Li, Z., Sun, Y., Zhu, J., Tang, S., Zhang, C., & Ma, H. (2021). Improve relation extraction with dual attention-guided graph convolutional networks. *Neural Computing and Applications*, *33*, 1773–1784.

Li, Z., Wallace, E., Shen, S., Lin, K., Keutzer, K., Klein, D., et al. (2020). Train big, then compress: Rethinking model size for efficient training and inference of transformers. In *International conference on machine learning* (pp. 5958–5968). PMLR.

Li, B., Yu, D., Ye, W., Zhang, J., & Zhang, S. (2023). Sequence generation with label augmentation for relation extraction. *Vol. 37*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 13043–13050).

Lin, Y., Liu, Z., Sun, M., Liu, Y., & Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. *Vol. 29*, In *Proceedings of the twenty-ninth AAAI conference on artificial intelligence* (pp. 2181–2187).

Liu, X., Ji, K., Fu, Y., Du, Z., Yang, Z., & Tang, J. (2021). P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. arXiv preprint arXiv:2110.07602.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., et al. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

Liu, C., Sun, W., Chao, W., & Che, W. (2013). Convolution neural network for relation extraction. In *International conference on advanced data mining and applications* (pp. 231–242). Springer.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, *55*(9), 1–35.

Liu, J., Zhang, Z., Guo, Z., Jin, L., Li, X., Wei, K., et al. (2023). KEPT: Knowledge enhanced prompt tuning for event causality identification. *Knowledge-Based Systems*, *259*, Article 110064.

Liu, X., Zheng, Y., Du, Z., Ding, M., Qian, Y., Yang, Z., et al. (2021). GPT understands, too. arXiv preprint arXiv:2103.10385.

Lu, Y., Liu, Q., Dai, D., Xiao, X., Lin, H., Han, X., et al. (2022). Unified structure generation for universal information extraction. arXiv preprint arXiv:2203.12277.

Miwa, M., & Bansal, M. (2016). End-to-end relation extraction using lstms on sequences and tree structures. arXiv preprint arXiv:1601.00770.

Mohajer, A., Daliri, M. S., Mirzaei, A., Ziaeddini, A., Nabipour, M., & Bavaghar, M. (2022). Heterogeneous computational resource allocation for NOMA: Toward green mobile edge-computing systems. *IEEE Transactions on Services Computing*, *16*(2), 1225–1238.

Mohajer, A., Sorouri, F., Mirzaei, A., Ziaeddini, A., Rad, K. J., & Bavaghar, M. (2022). Energy-aware hierarchical resource management and backhaul traffic optimization in heterogeneous cellular networks. *IEEE Systems Journal*, *16*(4), 5188–5199.

Mooney, R., & Bunescu, R. (2005). Subsequence kernels for relation extraction. *Advances in Neural Information Processing Systems*, *18*.

Nan, G., Guo, Z., Sekulić, I., & Lu, W. (2020). Reasoning with latent structure refinement for document-level relation extraction. In *Proceedings of the 58th annual meeting of the association for computational linguistics*.

Peng, N., Poon, H., Quirk, C., Toutanova, K., & Yih, W.-t. (2017). Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics*, *5*, 101–115.

Peters, M. E., Neumann, M., Logan IV, R. L., Schwartz, R., Joshi, V., Singh, S., et al. (2019). Knowledge enhanced contextual word representations. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing* (pp. 43–54).

Qian, L., Zhou, G., Kong, F., Zhu, Q., & Qian, P. (2008). Exploiting constituent dependencies for tree kernel-based semantic relation extraction. In *Proceedings of the 22nd international conference on computational linguistics* (pp. 697–704).

Quirk, C., & Poon, H. (2017). Distant supervision for relation extraction beyond the sentence boundary. In *Proceedings of the 15th conference of the European chapter of the association for computational linguistics* (pp. 1171–1182).

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, *1*(8), 9.

Robertson, S., Zaragoza, H., et al. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, *3*(4), 333–389.

Sainz, O., de Lacalle, O. L., Labaka, G., Barrena, A., & Agirre, E. (2021). Label verbalization and entailment for effective zero-and few-shot relation extraction. arXiv preprint arXiv:2109.03659.

Schick, T., & Schütze, H. (2020). It's not just size that matters: Small language models are also few-shot learners. arXiv preprint arXiv:2009.07118.

Schick, T., & Schütze, H. (2021). Exploiting cloze questions for few shot text classification and natural language inference. In *Proceedings of the 16th conference of the European chapter of the association for computational linguistics* (pp. 255–269).

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Vol. 39*, *Introduction to information retrieval*.

See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th annual meeting of the association for computational linguistics* (pp. 1073–1083).

She, H., Wu, B., Wang, B., & Chi, R. (2018). Distant supervision for relation extraction with hierarchical attention and entity descriptions. In *2018 international joint conference on neural networks* (pp. 1–8). IEEE.

Shen, Y., & Huang, X.-J. (2016). Attention-based convolutional neural network for semantic relation extraction. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers* (pp. 2526–2536).

Shin, T., Razeghi, Y., Logan, R. L., IV, Wallace, E., & Singh, S. (2020). Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 conference on empirical methods in natural language processing* (pp. 4222–4235).

Siino, M., Di Nuovo, E., Tinnirello, I., & La Cascia, M. (2022). Fake news spreaders detection: Sometimes attention is not all you need. *Information*, *13*(9), 426.

Soares, L. B., FitzGerald, N., Ling, J., & Kwiatkowski, T. (2019). Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th conference of the association for computational linguistics* (pp. 2895–2905).

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., et al. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing* (pp. 1631–1642).

Song, R., Liu, Z., Chen, X., An, H., Zhang, Z., Wang, X., et al. (2023). Label prompt for multi-label text classification. *Applied Intelligence*, *53*(8), 8761–8775.

Song, L., Zhang, Y., Wang, Z., & Gildea, D. (2018). N-ary relation extraction using graph state lstm. arXiv preprint arXiv:1808.09101.

Stoica, G., Platanios, E. A., & Póczos, B. (2021). Re-tacred: Addressing shortcomings of the tacred dataset. *Vol. 35*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 13843–13850).

Su, Y., Liu, H., Yavuz, S., Gur, I., Sun, H., & Yan, X. (2017). Global relation embedding for relation extraction. arXiv preprint arXiv:1704.05958.

Suchanek, F. M., Ifrim, G., & Weikum, G. (2006). Combining linguistic and statistical analysis to extract relations from web documents. In *Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 712–717).

Tang, H., Cao, Y., Zhang, Z., Cao, J., Fang, F., Wang, S., et al. (2020). Hin: Hierarchical inference network for document-level relation extraction. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 197–209).

Tian, Y., Song, Y., & Xia, F. (2022). Improving relation extraction through syntax-induced pre-training with dependency masking. In *Findings of the association for computational linguistics: ACL 2022* (pp. 1875–1886).

Vashishth, S., Joshi, R., Prayaga, S. S., Bhattacharyya, C., & Talukdar, P. (2018). Reside: Improving distantly-supervised neural relation extraction using side information. arXiv preprint arXiv:1812.04361.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Wang, H., Focke, C., Sylvester, R., Mishra, N., & Wang, W. (2019). Fine-tune bert for docred with two-step process. arXiv preprint arXiv:1909.11898.

Wang, S., Xu, Y., Fang, Y., Liu, Y., Sun, S., Xu, R., et al. (2022). Training data is more valuable than you think: A simple and effective method by retrieving from training data. In *Proceedings of the 60th annual meeting of the association for computational linguistics* (pp. 3170–3179).

Wu, S., & He, Y. (2019). Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 2361–2364).

Xiao, M., & Liu, C. (2016). Semantic relation classification via hierarchical recurrent neural network with attention. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers* (pp. 1254–1263).

Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., & Jin, Z. (2015). Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1785–1794).

Xue, F., Sun, A., Zhang, H., & Chng, E. S. (2021). Gdpnet: Refining latent multi-view graph for relation extraction. *Vol. 35*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 14194–14202).

Yamada, I., Asai, A., Shindo, H., Takeda, H., & Matsumoto, Y. (2020). LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 conference on empirical methods in natural language processing* (pp. 6442–6454).

Zeng, Y., Li, Z., Chen, Z., & Ma, H. (2023). Aspect-level sentiment analysis based on semantic heterogeneous graph convolutional network. *Frontiers of Computer Science*, *17*(6), Article 176340.

Zeng, Y., Li, Z., Tang, Z., Chen, Z., & Ma, H. (2023). Heterogeneous graph convolution based on in-domain self-supervision for multimodal sentiment analysis. *Expert Systems with Applications*, *213*, Article 119240.

Zeng, D., Liu, K., Chen, Y., & Zhao, J. (2015). Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1753–1762).

Zhang, D., & Wang, D. (2015). Relation classification via recurrent neural network. arXiv preprint arXiv:1508.01006.

Zhang, Z., Yu, B., Shu, X., Mengge, X., Liu, T., & Guo, L. (2021). From what to why: Improving relation extraction with rationale graph. In *Findings of the association for computational linguistics: ACL-IJCNLP 2021* (pp. 86–95).

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. *Vol. 28*, In *Human centered computing - 5th international conference* (pp. 560–567).

Zhang, S., Zheng, D., Hu, X., & Yang, M. (2015). Bidirectional long short-term memory networks for relation classification. In *Proceedings of the 29th Pacific Asia conference on language, information and computation* (pp. 73–78).

Zhang, Y., Zhong, V., Chen, D., Angeli, G., & Manning, C. D. (2017). Position-aware attention and supervised data improve slot filling. In *Conference on empirical methods in natural language processing* (pp. 35–45).

Zhong, W., Gao, Y., Ding, N., Qin, Y., Liu, Z., Zhou, M., et al. (2022). ProQA: Structural prompt-based pre-training for unified question answering. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics* (pp. 4230–4243).

Zhou, W., & Chen, M. (2021). An improved baseline for sentence-level relation extraction. arXiv preprint arXiv:2102.01373.

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., et al. (2016). Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: short papers)* (pp. 207–212).

Zhou, G., Zhang, M., Ji, D., & Zhu, Q. (2007). Tree kernel-based relation extraction with context-sensitive structured parse tree information. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning* EMNLP-coNLL, (pp. 728–736).